

Project Assignment



Cybersecurity Professional Program
Introduction to Python
for Security

Final Project

PY-07-L1

ARP Spoof Detector



Project Objective

Implement learned skills to create an automated program that can be used in real-life scenarios.



Project Mission

Create a program that can detect active ARP Spoofing attacks on host machines.



Project Duration

2–4 hours



Requirements

- Working knowledge of Python
- Working knowledge of Python's OS module



Resources

- Environment & Tools
 - Windows, macOS, and Linux
 - PyCharm
 - Python 3
 - Kali Linux ISO

Project Scenario

In the past few days, the HackRS company has been experiencing network connection issues, such as low bandwidth and disconnections. The IT team couldn't find the cause of the issue, and the company decided to ask for your assistance.

You suspect the network is under an On-Path attack. As a Python script expert, you need to create a script that automatically identifies ARP Spoofing behavior on workstations.

In the first stage, the script should read a station's ARP table and extract the address from it. The script should search for MAC duplications in table entries and log every ARP Spoofing event in the next stage.

Project Task 1: ARP Table Extraction

The first part of the program must handle the extraction of the ARP table data. The data must be saved in a dynamic structure in which the IP address corresponds to the MAC address. The structure is required for later comparison operations to identify duplicated addresses.

- 1** Plan a function that will extract the ARP table data from the machine. How can Python access this type of data? How should the data be saved for later use?
- 2** Import the required modules for the program.
- 3** Define a function that will handle the ARP table data extraction.
- 4** Create three variables: one to store the ARP table data, another to store a list of the separated lines, and the third to store the final filtered data.
- 5** Iterate over the lines and save the IP addresses and corresponding MAC addresses after data filtration. Only IP and MAC addresses should be saved in the third variable. Filter the rest of the data, such as the interface's IP address or broadcast data.

Project Task 2: Identifying MAC Address Duplication

The second part of the program must take the extracted data and identify if an ARP Spoofing attack is underway. This is done by locating MAC address duplications in the ARP table entries.

- 1** Plan a function that will identify MAC address duplication.
- 2** Define a function to identify duplication in MAC addresses. The function should accept a parameter.
- 3** Create a variable to store iterated MAC addresses for later comparison.
- 4** Iterate over the recorded MAC addresses and compare them to the saved ones to identify duplications. Print a message that notifies when duplication is identified.
- 5** The ARP extracting function passes the filtered data to the current function.

Project Task 3: Logging Events

The third part of the program will handle the logging of the ARP Spoof activity.

- 1** Plan a function that will log every ARP Spoofing event and save it to a file.
- 2** Define a function to handle ARP Spoofing event logging. The function should accept data regarding the event.
- 3** Create a variable to store the date and time of the event.
- 4** Save the logged data to a file.
- 5** The ARP Spoof identification function passes a message to the log creator function.
- 6** Add execution control to ensure the program is executed only if its file is directly executed.

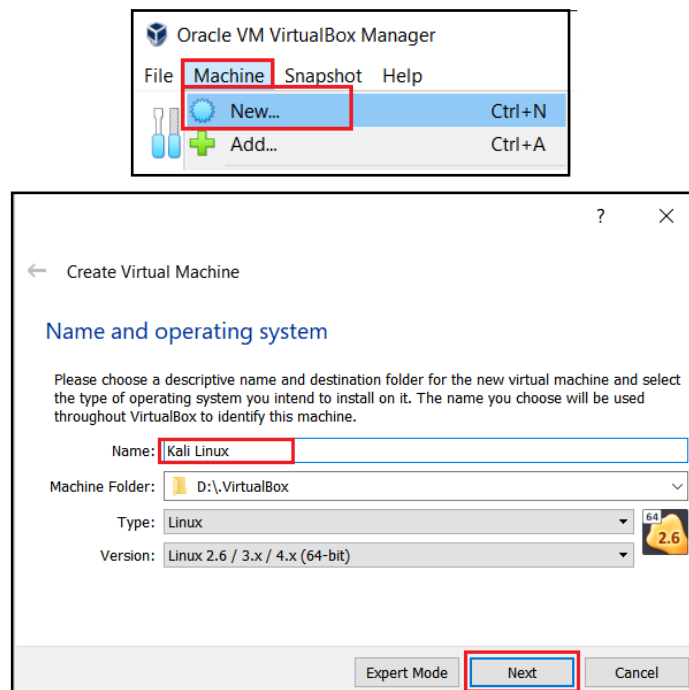
Project Task 4: Testing

This task is the final step in the project. The program to identify ARP Spoofing activity in the network is complete. The last step is to test its functionality.

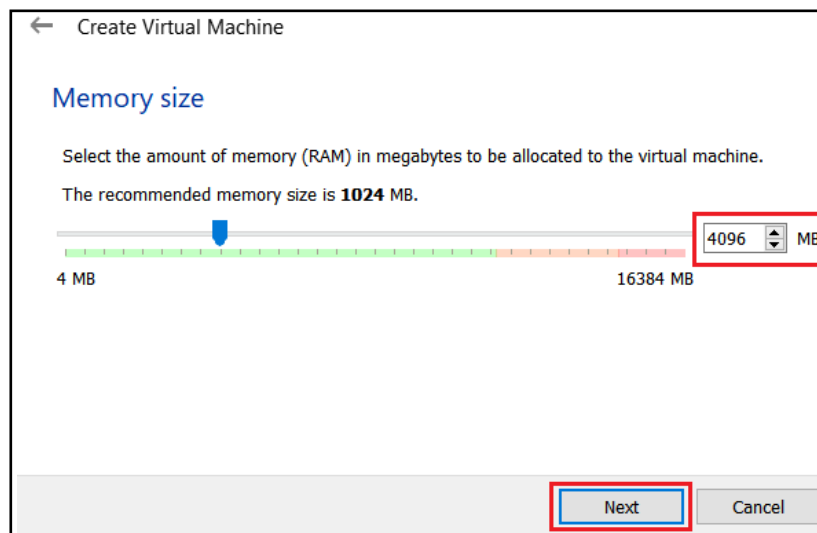
To check the script, an ARP Spoof attack must be launched. This can be done from a Kali Linux VM against a Windows host or virtual machine.

The OS should be configured to use the Bridge Adapter network configuration for direct communication with the router.

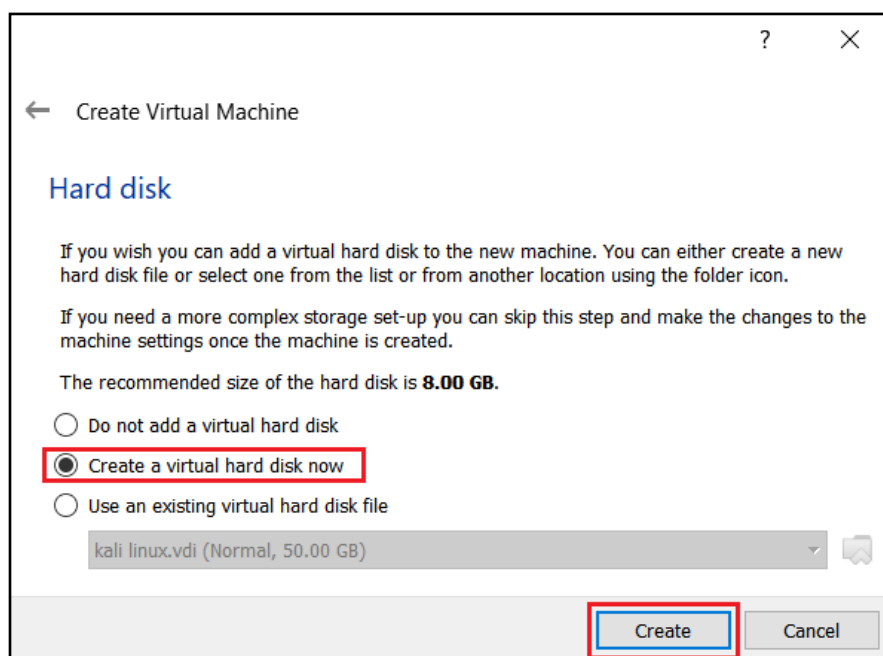
- 1 Open VirtualBox, select the **Machine** tab, click **New...**, and name the VM **Kali Linux**.



- 2 Set the memory to **2048 MB** for proper functionality. For enhanced functionality, you can use a higher setting (per the computer's available resources).



- 3 Select **Create a virtual hard disk now** in the next window and click **Create**.



4 Select **VHD** for the hard disk file type and click **Next**.

← Create Virtual Hard Disk

Hard disk file type

Please choose the type of file that you would like to use for the new virtual hard disk. If you do not need to use it with other virtualization software you can leave this setting unchanged.

☐ VDI (VirtualBox Disk Image)

☒ VHD (Virtual Hard Disk)

☐ VMDK (Virtual Machine Disk)

Expert Mode Next Cancel

5 Select **Dynamically allocated** and click **Next**.

← Create Virtual Hard Disk

Storage on physical hard disk

Please choose whether the new virtual hard disk file should grow as it is used (dynamically allocated) or if it should be created at its maximum size (fixed size).

A **dynamically allocated** hard disk file will only use space on your physical hard disk as it fills up (up to a maximum **fixed size**), although it will not shrink again automatically when space on it is freed.

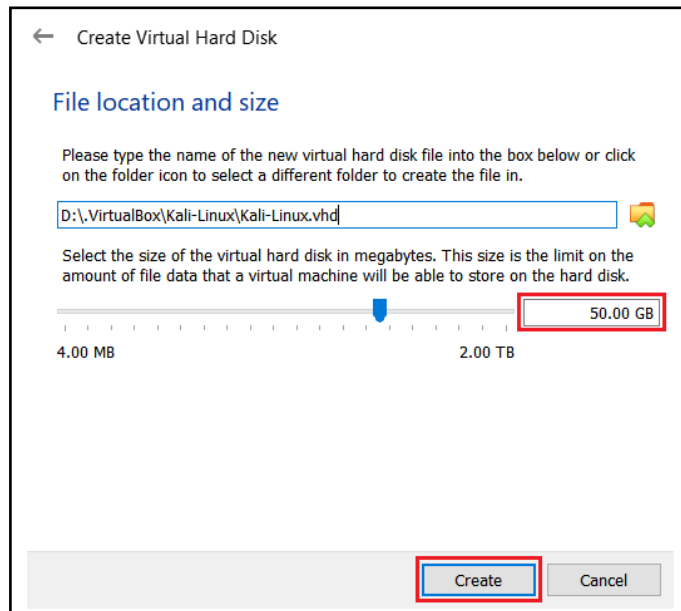
A **fixed size** hard disk file may take longer to create on some systems but is often faster to use.

☒ Dynamically allocated

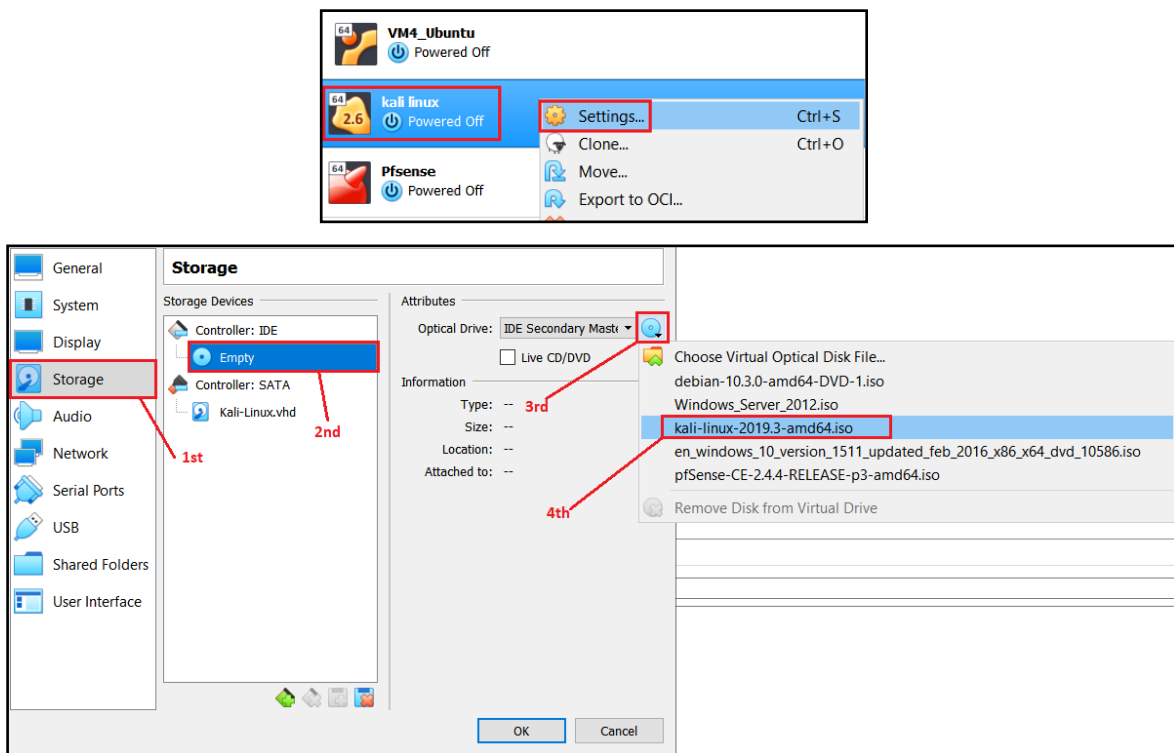
☐ Fixed size

Next Cancel

- 6 Select the file location and set the storage size to **50 GB**. If your computer does not have enough storage, you can set it to **20 GB**.



- 7 Right-click the Kali VM, select **Settings...**, and insert the **Kali Linux ISO** file into the drive.



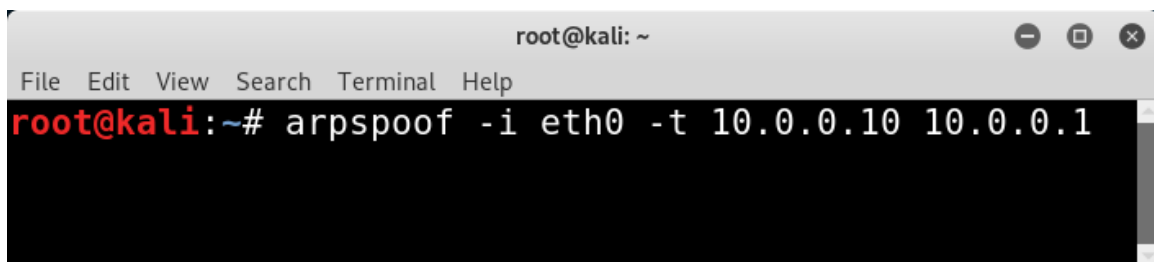
- 8 Run the virtual machine and select the live version.



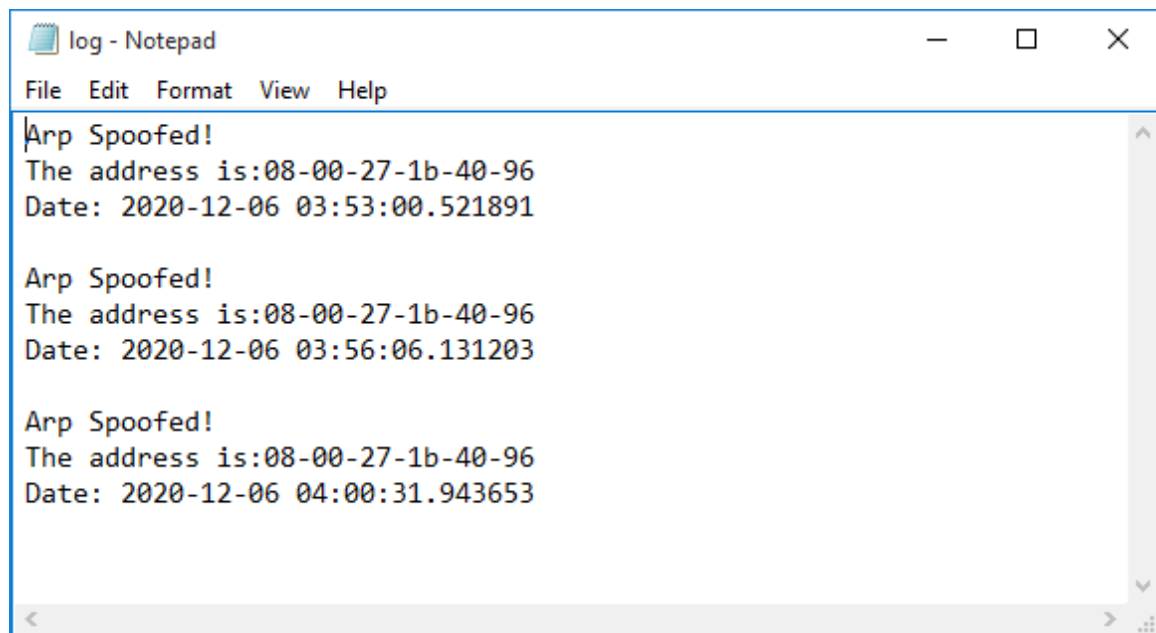
- 9 Open the terminal by clicking on it in the options on the left.



- 10 Run the ***arpspoof -i eth0 -t <target IP> <spoofed IP>*** command.



11 Execute the script and note that a log was created.



```
log - Notepad
File Edit Format View Help
Arp Spoofed!
The address is:08-00-27-1b-40-96
Date: 2020-12-06 03:53:00.521891

Arp Spoofed!
The address is:08-00-27-1b-40-96
Date: 2020-12-06 03:56:06.131203

Arp Spoofed!
The address is:08-00-27-1b-40-96
Date: 2020-12-06 04:00:31.943653
```